

Final Report - NAG2-1107
A Linux Workstation for High Performance Graphics

Robert Geist, PI
James Westall, co-PI

Submitted to
NASA Ames Research Center
University Affairs Office
Moffett Field, CA 94035

July 11, 2000

JUL 19 2000

cc: 202A-3 v

1 Project Overview.

The primary goal of this effort was to provide a low-cost method of obtaining high-performance 3-D graphics using an industry standard library (OpenGL) on PC class computers. Previously, users interested in doing substantial visualization or graphical manipulation were constrained to using specialized, custom hardware most often found in computers from Silicon Graphics (SGI). We provided an alternative to expensive SGI hardware by taking advantage of third-party, 3-D graphics accelerators that have now become available at very affordable prices. To make use of this hardware our goal was to provide a free, redistributable, and fully-compatible OpenGL work-alike library so that existing bodies of code could simply be recompiled for PC class machines running a free version of Unix. This should allow substantial cost savings while greatly expanding the population of people with access to a serious graphics development and viewing environment. This should offer a means for NASA to provide a spectrum of graphics performance to its scientists, supplying high-end specialized SGI hardware for high-performance visualization while fulfilling the requirements of medium and lower performance applications with generic, off-the-shelf components and still maintaining compatibility between the two.

2 Project Description.

2.1 Task 1.

Task 1, begun in January of 1997, called for the design and implementation of a kernel-side driver for a graphics board based on 3DLabs' GLINT 500TX and DELTA chips. Version 1.0 was delivered to NASA in the summer of 1997. The current version, 4.3, contains numerous modifications and improvements suggested by Ames. It runs under Linux 2.2.5 on a dual-processor, Micron Millenia Pro2 (dual PPro 200MHz) equipped with an Omnicomp Pro88 graphics board containing the 500TX and DELTA chips. Hereafter we refer to this hardware system as the *GLINT system*.

The board driver is a dynamically loadable kernel module, and it requires no kernel modifications whatsoever. Thus the driver may be extracted, modified, and re-installed without system re-boot. The driver treats the board as a standard Linux character device and uses the *open* and *ioctl* system calls to map both FIFO and DMA control to user space. Thereafter, the user process can load dma buffers and send

them directly to the board with minimal kernel overhead. Staying out of the kernel avoids performance bottlenecks.

2.2 Task 2.

Task 2 called for the design and implementation of the client side Mesa (OpenGL-lookalike) library for the GLINT system. The initial version, 1.0, was completed in December, 1997. A later version, 2.0, was a drop-in for Mesa 3.0 beta 5 and used calls to the Task 1 board driver to access the following features in hardware:

- fast clear framebuffer
- fast clear depthbuffer
- z buffering
- double buffering
- smooth-shaded triangles, lines, meshes
- texture mapping
- alpha blending
- windowing support
- scissoring (screen)
- clipping (window)
- framebuffer write masks

Through this package, hardware-assisted, OpenGL rendering became available in windows of user-specified size and screen location.

2.3 Task 3.

Task 3 called for the design and implementation of an X11R6 server for the GLINT system. Version 1.0 of this X server was delivered to NASA, along with then-current versions of the OpenGL (Mesa) driver and the board driver, in August of 1998. The X server calls the board driver to memory-map the framebuffer and then uses direct writes to the framebuffer for most operations. The exceptions are the *CopyWindow* and *CopyArea* functions, where direct writes would be too slow. Here the X server uses the fast copy operations available through direct loads to the 500TX-DELTA graphics registers.

Synchronization between the OpenGL driver and the X server was, at this stage, minimal. Since both the X server and the OpenGL application were required to open the graphics board, the board driver maintained counters that allowed proper device initialization and shutdown. The *aux/tk* library call, *auxInitWindow()*, opened the requested X window and passed coordinates through to the OpenGL driver so that graphics could be rendered into the window at full board speed. Since the OpenGL driver used exclusively DMA access to the board, and the X server never used DMA access, conflicts arising through different rendering contexts could be avoided.

Driver code was completely transparent at the user level. As an example, the following user level code could be compiled and executed as-is:

```

/* OpenGL render of single, smooth-shaded triangle */
#include <stdio.h>
#include "GL/gl.h"

void zdisplay(void);

main()
{
    auxInitPosition(256,128,512,512);
    auxInitWindow("mywindow");
    auxReshapeFunc(0);
    auxMainLoop(zdisplay);
}

void zdisplay(void)
{
    glShadeModel(GL_SMOOTH);
    glDrawBuffer(GL_FRONT);
    glClearColor(0.9,0.8,0.9,1.0);
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_TRIANGLES);
        glColor3f(1.0,0.0,0.0);
        glVertex3f(-0.8,0.6,0.0);
        glColor3f(0.0,1.0,0.0);
        glVertex3f(0.7,0.0,0.0);
        glColor3f(0.0,0.0,1.0);
        glVertex3f(0.1,-0.6,0.0);
    glEnd();
    glFlush();
}

```

As an output example, the color image of figure 1 is a screen capture from the GLINT system, taken while rendering the classic *teapots in the fog* OpenGL demo.

Throughout the project, graphics performance has been a major focus of the design. Performance of the GLINT system continues to compare favorably with that of an (R10000-based) SGI O2 on both our own “bounce” benchmark (lighting and fill emphasis) and the CDRS “viewperf” benchmark (geometry emphasis). The O2 was purchased at the same time as the GLINT system (5/97) and had a list price of \$23,685, with a University-discounted price of approximately \$18,000.

The relatively weak floating point performance of the 200MHz Pentium Pro continues to be a limiting factor. We conjecture that an upgrade to a 750MHz Pentium III AGP would yield overall graphics performance that substantially exceeds that of a high-end SGI O2, even if the PC were equipped with the new low-end GLINT card, the Permedia III.

2.4 Task 4.

Task 4, completed in February, 2000, provided full OpenGL/X synchronization, including the capability of allowing multiple, simultaneous, independent OpenGL applications to use direct-rendering into individual X windows.

To support multiple, simultaneous applications, within the same user process or separate processes,

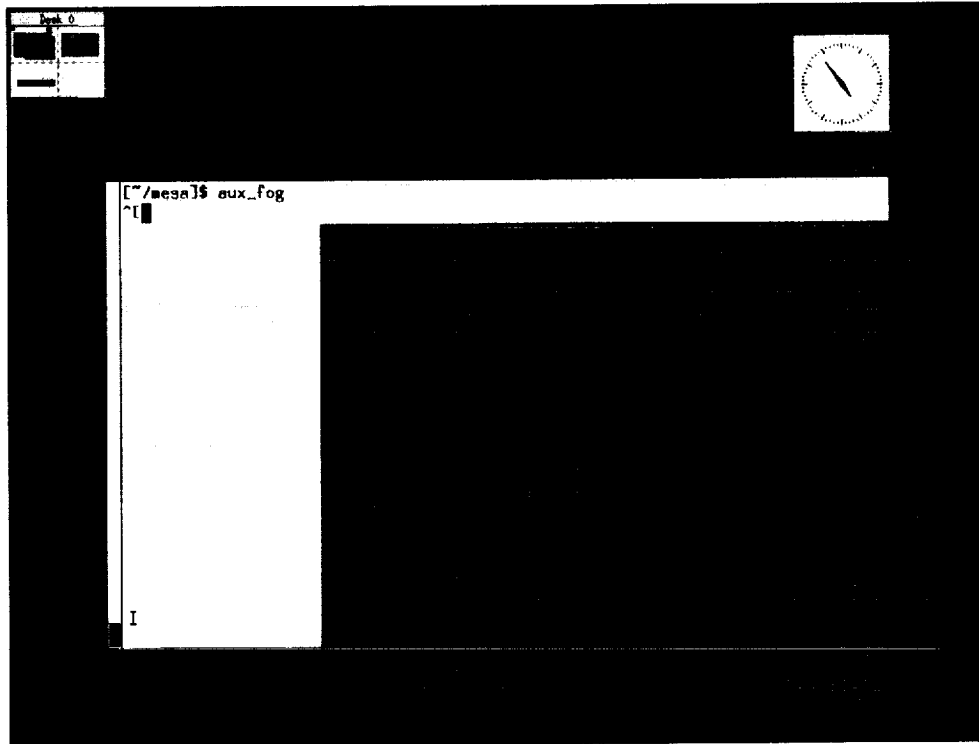


Figure 1: Screen capture from GLINT system.

graphics rendering contexts must be maintained. Nevertheless, an overriding design constraint of our system was that changes to the static OS kernel must be limited to the export of symbols for use by our board driver module. Thus, we could not add process table information or identify processes as “graphics” processes. Instead, since all applications render through the DMA pipe, we introduced coordination at the point of access to the DMA buffer pool.

We attach a rendering context id to each DMA buffer (header). Before each DMA buffer is transmitted to the board, its context id is checked against the current context id. If they differ, a rendering context switch is invoked through direct writes to the graphics registers. The board driver maintains the separate rendering contexts, and the X window id serves as context id. Rendering contexts are generated and made current through the Mesa driver commands, *GlintMesaCreateContext()* and *GlintMesaMakeCurrent()*.

Synchronization with the X server represents a difficult problem. The X server is small and portable largely because it makes use of direct writes to the frame buffer for most operations. This allows generic code from the X source tree to be used for most updates. Only a small number of performance-sensitive routines have been re-written specifically for the 500TX-DELTA.

Direct writes to the framebuffer or to the graphics registers while DMA transfer is in progress will produce unpredictable results. Fortunately, server event handling occurs at a single point, in the *dispatch()* routine of *Xserver/dix/dispatch.c*. At this point, the server can await completion of the current DMA transfer/render, if one is in progress, and then handle pending X events under the X rendering context in a non-preemptable fashion. Suspension of the server (within the *dispatch()* routine) then allows DMA transfers to resume, with rendering contexts switched in as needed.

Rendering contexts can be minimized, and so performance enhanced, if preemption is allowed only on frame boundaries. Since multiple buffers may contribute to a single frame, this can degrade the responsive-

ness of the X server, which must always await a frame boundary. Nevertheless, the performance trade-off is balanced by giving the X server priority over all applications at each frame boundary.

The user level windowing and event interface was moved from the somewhat dated *aux/tk* library to the newer *glut* library that is part of Mesa 3.0 beta 5. All mapping of OpenGL commands is now through glX (Mesa-3.0/X/fakeglx.c). This gives greater portability of OpenGL applications. Applications developed on SGI workstations can be compiled and executed without modification.

We also provided a complete system implementation for the GLINT Permedia II chipset, which offers a spectacular price/performance point in graphics hardware. Performance substantially exceeds that of the 500TX-DELTA chipset at substantially lower cost. The Permedia II register set differs from that of the 500TX-DELTA, and our Permedia II system package differs in some minor ways from the system package for the 500TX-DELTA. An extension to the Permedia III chipset is underway, but it is unsupported.

3 Publications.

Several publications resulted directly from this grant; they include:

- Arens, A., "Multiple Application OpenGL (Mesa) Support on GLINT Boards," *Proc. 38th Annual ACM Southeast Conf.*, Clemson, South Carolina, April, 2000, pp. 26 - 35.
- Westall, J., "Accelerated 3-D Graphics in an XFree86 Environment," *Proc. 38th Annual ACM Southeast Conf.*, Clemson, South Carolina, April, 2000, pp. 36 - 37.
- Geist, R., Treglia, D., and Westall, J., "Real-time, 3-D Graphics in Linux," *Proc. 5th Annual Linux Expo*, Raleigh, North Carolina, May, 1999, pp. 9 - 17.
- Geist, R., Westall, J., Treglia, D., and Smotherman, M., "Real-time, 3-D Graphics for the Linux PC," *Proc. of the 24th Annual Int. Conf. of the Computer Measurement Group (CMG98)*, Anaheim, California, December, 1998, pp. 863 - 873.
- Geist, R., and Westall, J., "Bringing the High-End to the Low-End: High Performance Device Drivers for the Linux PC," *Proc. of the 36th Annual ACM Southeast Conf.*, Atlanta, Georgia, April, 1998, pp. 251 - 260.

It is worth noting that the first-named paper, the M.S. thesis of graduate student Andreas Arens, won first prize in the ACM Southeast Student Paper Competition. Authors Arens and Treglia were graduate students supported by this grant.

4 Code Availability.

Source code and installation instructions are available through NASA contact Lonhyn Jasinskyj, lonhyn-nas.nasa.gov.

5 Subject Inventions.

There were no inventions suitable for patent by any of the grant participants during the entire grant period.